# USB-IR-Boy, Linux Remote Receiver

Written By: nuess0r

## PARTS:

- FREESCALE MC68HC908JB8JPE 8BIT MCU 8K FLASH USB Farnell 1148404 (1)
- MULTICOMP IC SOCKET DIL 0.3" 20WAY Farnell 1103848 (1)
- VISHAY TSOP34836 PHOTODIODE IR RECEIVER 36KHZ Farnell 4913127 (1)
- AVX CRYSTAL HC-49 6MHZ Farnell 1368787 (1)
- USB Cable is from a broken USB Mouse. (1)
- Case is from a HP IRDA receiver (HP C4103A) for HP printers. (1)
- MOhm Resistor (10)
- nF Capacitor (100)
  *2*
- pF Capacitor (22)
  *2*

## SUMMARY

One day I realized that I would like to have the ability to control my computer's media player (for me this is Amarok under Linux) with the IR remote from my NAD amplifier.

### Description

This project describes an infrared (IR) receiver for IR remote controls commonly used by consumer electronic devices like TVs, amplifiers, DVD players, etc. Most of them use the same coding scheme, called RC-5, so you can use your extra remotes or some unused buttons on your remote together with this receiver.

As the name suggests, this receiver is connected to the USB port of your computer.

This receiver can be used with the well-known Linux software LIRC. There is a description of my LIRC/Amarok setup below.

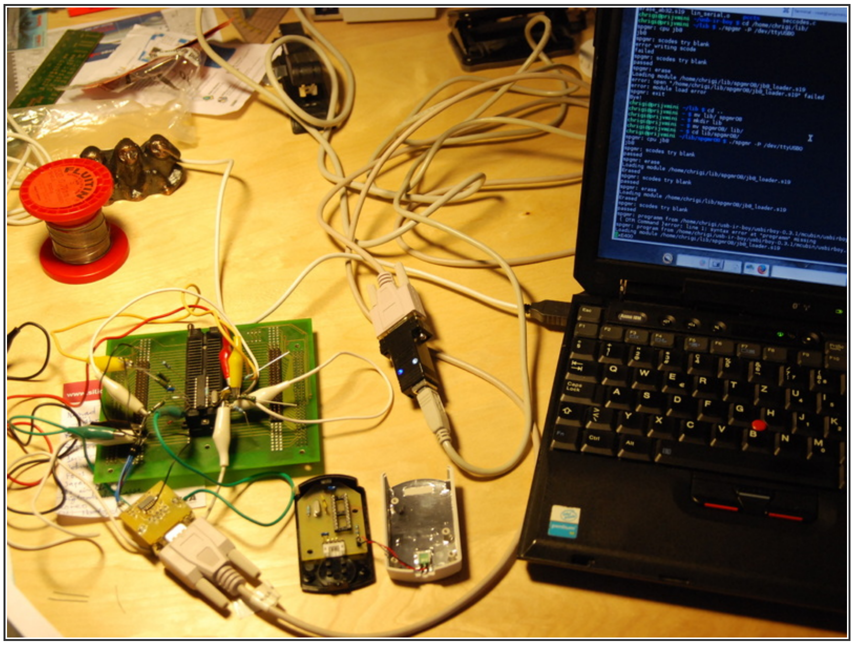
I found the original of this project and decided that I had to build one for myself.

It's a Sourceforge project and I tried to send my stuff to the original developer without success. So I am publishing it here.

## Step 1 — Get all needed stuff

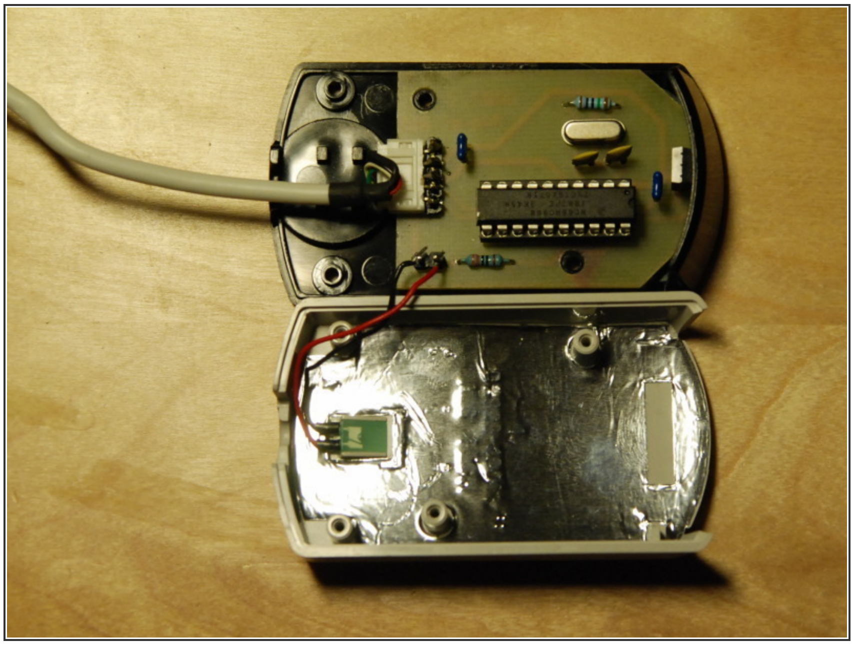
- Order the parts you need. In the parts list are Farnell/Newark order numbers.
- Make the PCB. At the moment there is no fabricated PCB available; you have to do it on your own. You can also build the circuit onto a prototype board since there are only a few components to wire.
- Get a small enclosure. There has to be something like a "window" in front of it. Place the IR receiver behind this. You can also simply cut out a hole for the receiver.
- If you use the same type of IRDA receiver box as I did, remove the rubber feet on the bottom and keep them. Under the feet you will see the case screws. Open the case, cut the wires to the LED close to the board and remove the PCB. We don't need it.

## Step 2 — Get the firmware and program the microcontroller



- The most complicated part is to get your microcontroller programmed. If you are lucky you know someone with a programmer for Freescale HC8 controllers. Most of us will not...
- Build a programmer. For this refer to the original project description. It is rather clear and complete.
- Download all necessary files: USB-IR-Boy Firmware, M68HC908 Programmer Utility (or install it through your package manager).
- Flash the firmware to the microcontroller: This part is well documented in the original project docs, chapter *Programming the board MCU*.

## Step 3 — Build the receiver circuit



- Solder the few parts to the PCB. Use the attached file **usb-ir-boy-SilkSCmp.pdf** as an assembly drawing. Start with the resistor, followed by the IC socket and the crystal, then the capacitors and finally the IR receiver.
- If you have one at hand (or desoldered from the mouse where you got the USB cable), solder in a connector for the USB cable. Else, solder the cable directly onto the PCB. USB D+ is the green wire, D- is the white one.
- Put the programmed microcontroller into the socket.
- Solder the LED wires to the board. We use it for testing only (because the firmware is only using it as a heartbeat, not as a receiving notification).
- Plug the unit into a USB port. The LED should blink in a constant rhythm.
- Unsolder the LED and close the case.

## Step 4 — See if it works and install Linux driver

- Let's see if our device is talking to the USB port: `lsusb -v -d fffe:0000`
- This should return something like: `idVendor 0xfffe idProduct 0x0000 bcdDevice 0.31 iManufacturer 2 I made it! iProduct 1 IR BOY iSerial 0 bNumConfigurations 1`
- Install the Linux Kernel module according to the original project documentation, Chapter *Kernel module*.

## Step 5 — Configure LIRC and Amarok

- Install LIRC and the LIRC tools using your package manager. For general help with LIRC visit their homepage: http://lirc.org/
- I had some problems with irrecord. It recognised the remote control in raw mode and not as something more standard. The produced `*.conf` file was not working.
- Using a `*.conf` file for a similar remote control provided by the LIRC package is working fine for me. It seems that this is a known problem in the latest firmware version. On the mailing list Norbert Hohenbichler has reported this error.
- To use Amarok with your freshly built USB-IR-Boy and your running LIRC daemon I used this description. In short, make sure lircd and irexec are running, and configure your .lirc configuration in such a way that for a key press irexec is executed and sends a DCOP message to Amarok to do what you want. In this way you can use your remote control for all DCOP-capable software!
- One example section of my .lirc file: `begin prog = irexec remote = nad450 button = TapeA-Play repeat = 2 config = dcop amarok player volumeUp end`

This document was last generated on 2012-11-02 07:39:51 AM.